

Title	0-1 整数変数を含む非凸2次最適化問題の非負半正定値緩和に対する面的縮小と効率的解法 (最適化手法の深化と広がり)
Author(s)	田中, 未来; 中田, 和秀; 脇, 隼人
Citation	数理解析研究所講究録 (2012), 1773: 186-197
Issue Date	2012-01
URL	http://hdl.handle.net/2433/171701
Right	
Type	Departmental Bulletin Paper
Textversion	publisher

0-1 整数変数を含む非凸 2 次最適化問題の 非負半正定値緩和に対する面的縮小と効率的解法*

田中未来†

中田和秀‡

脇隼人§

概要

本論文では 0-1 整数変数を含む非凸 2 次最適化問題に対する非負半正定値緩和問題を効率よく解く方法を提案する. 始めに, 非負半正定値緩和問題が退化していることを用いて問題を縮小する方法を提案する. この縮小は面的縮小の部分的な適用とみなすことができるため, 主双対パス追跡法の安定性が高まることが期待できる. 続いて, 縮小後の問題を主双対パス追跡法で解く. 探索方向の計算では, 大規模な線形方程式系を Krylov 部分空間法の一つである PSQMR 法を用いて解くことで効率化を図る. さらに, PSQMR 法の収束性を高めるために 3 つの前処理を提案する. 最後に, 数値実験により提案手法の効率性を示す.

1 はじめに

本論文では次の 0-1 整数変数を含む非凸 2 次最適化問題について考える:

$$\begin{array}{ll} \text{minimize} & \mathbf{x}^T \mathbf{\Gamma} \mathbf{x} + 2\mathbf{\gamma}^T \mathbf{x} \\ \text{subject to} & \alpha_i^T \mathbf{x} = \beta_i \quad (i = 1, \dots, m), \\ & \mathbf{x} \geq \mathbf{0}, \\ & x_j \in \{0, 1\} \quad (j \in B). \end{array} \quad (\text{MBNQO})$$

ここで $\mathbf{x} \in \mathbb{R}^n$ は決定変数, $\mathbf{A} = (\alpha_1, \dots, \alpha_m)^T \in \mathbb{R}^{m \times n}$, $\boldsymbol{\beta} = (\beta_1, \dots, \beta_m)^T \in \mathbb{R}^m$, $\boldsymbol{\gamma} \in \mathbb{R}^n$, $\mathbf{\Gamma} \in \mathcal{S}^n$, $B \subset \{1, \dots, n\}$ は与えられたデータである. なお, \mathcal{S}^n は n 次対称行列の空間を表す. この問題は幅広い応用範囲をもつ [2, 7]. しかし, この問題は離散構造と非凸性を有する問題であり, (MBNQO) の大域的最適解を求めることは NP 困難である. この問題に対して Burer [7] は, 比較的ゆるい仮定の下で (MBNQO) の最適値が次の完全正値最適化問題の最適値と一致することを示している:

$$\begin{array}{ll} \text{minimize} & \mathbf{\Gamma} \bullet \mathbf{X} + 2\mathbf{\gamma}^T \mathbf{x} \\ \text{subject to} & \alpha_i^T \mathbf{x} = \beta_i \quad (i = 1, \dots, m), \\ & \alpha_i^T \mathbf{X} \alpha_i = \beta_i^2 \quad (i = 1, \dots, m), \\ & x_j = X_{jj} \quad (j \in B), \\ & \begin{pmatrix} 1 & \mathbf{x}^T \\ \mathbf{x} & \mathbf{X} \end{pmatrix} \in (C^{1+n})^*. \end{array} \quad (\text{CPO})$$

* 本研究は科研費 (22710136), 科研費 (22740056) の助成を受けたものである.

† 東京工業大学 大学院社会理工学研究科 経営工学専攻

‡ 東京工業大学 大学院社会理工学研究科 経営工学専攻

§ 電気通信大学 大学院情報理工学研究科 情報・通信工学専攻

ここで $x \in \mathbb{R}^n$, $X \in S^n$ は決定変数, \bullet は行列の内積, すなわち $\Gamma \bullet X = \text{tr}(\Gamma^T X)$ を表し, $(C^n)^*$ は n 次完全正値行列のなす錐 [1, 4] を表す. (CPO) は閉凸錐上の線形最適化問題だが, 最適解を求めることは依然として難しい. 実際, 与えられた行列が完全正値かどうかを判定することは NP 困難であることが Dickinson and Gijben [9] によって示されている.

Burer [7] は (CPO) の最適値に対する強い下界を求めるために非負半正定値緩和を提案している. (CPO) に対する非負半正定値緩和問題は (CPO) における完全正値制約 $(1, x^T; x, X) \in (C^{1+n})^*$ を非負半正定値制約 $(1, x^T; x, X) \in S_+^{1+n} \cap S_{(+)}^{1+n}$ で置き換えることによって得られる. ここで S_+^n と $S_{(+)}^n$ はそれぞれ n 次半正定値対称行列のなす錐と n 次非負対称行列のなす錐であり,

$$(A, B; C, D) = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

である. この非負半正定値緩和は単純な半正定値緩和よりも強い緩和となることが Ge and Ye [12] によって指摘されている. また, 非負半正定値緩和問題は半正定値最適化問題に帰着することができるため, SeDuMi [20], SDPA [27], SDPT3 [24] のような半正定値最適化問題に対する既存のソルバを用いて解くことができる.

しかし, このアプローチには次の 2 つの問題点がある. 1 つめは, 非負半正定値緩和問題は実行可能内点解をもたないため, 主双対内点法の数値的安定性が悪くなることである. 2 つめは, 中規模の非負半正定値緩和問題であっても, 等価な半正定値最適化問題が非常に大規模な問題となることである. 実際, 非負半正定値緩和問題と等価な半正定値最適化問題は $(n+1)(n+2)/2 + 2m + |B| + 1$ 個の線形等式制約をもつため, 通常の主双対内点法の各反復では $(n+1)(n+2)/2 + 2m + |B| + 1$ 次の線形方程式系を解くことになる. 本論文ではこれら 2 つの問題点を解決する方法を提案する.

提案手法では, はじめに非負半正定値最適化問題が実行可能内点解をもたないことを用いて問題のサイズを縮小する. この縮小法は, 面的縮小 [5, 6, 17, 18, 19, 26] の部分的な適用とみなすことができる. 本論文で提案する縮小法は不完全な面的縮小であるため, 実行可能内点解を持つとは限らないが, 主双対内点法の安定性が向上することが期待できる.

縮小後の非負半正定値緩和問題であっても等価な半正定値最適化問題は大規模になるため, 通常の主双対内点法を用いて解く際には膨大な計算時間を要する. 主双対内点法において特に計算時間のかかる計算は探索方向の計算である. 実際, 縮小後の非負半正定値緩和問題と等価な半正定値最適化問題は $(n+1)(n+2)/2 + |B| + 1$ 本の線形等式制約を持つため, 各反復で $(n+1)(n+2)/2 + |B| + 1$ 次元の線形方程式系を解く必要がある.

さらに本論文では, 縮小後の非負半正定値緩和問題を効率よく解く方法を提案する. 提案手法では Toh [23] によって提案された凸 2 次半正定値最適化問題に対する主双対パス追跡法に基づき, 縮小後の非負半正定値緩和問題を半正定値最適化問題に帰着させることなく主双対パス追跡法で解く. 各反復における探索方向の計算では, $(n-m+1)^2 + |B| + 1$ 次元の線形方程式系を Krylov 部分空間法的一种である PSQMR 法 [11] を用いて解く. Krylov 部分空間法の収束性は悪条件な線形方程式系を解く場合に悪化することが知られている. 今回の場合, 主双対パス追跡法の反復点が最適解に近づくにつれて解くべき線形方程式系の係数行列は悪条件になる. そこで本研究では, PSQMR 法の収束性を高めるために Toh [23] によって提案された 3 つの前処理を今回の場合に拡張する.

本論文は次のような構成からなる. 2 節では, 非負半正定値緩和問題のサイズを縮小する方法を提案する. 3 節では, 縮小後の問題に対する主双対パス追跡法を提案する. 4 節では, 提案手法の効率性を示す数値実験の結果を示す. 最後に 5 節で総括を行う.

以下, 本論文で用いる記号の定義を行う. 2 つの行列 $P \in \mathbb{R}^{n \times n}$, $Q \in \mathbb{R}^{m \times m}$ に対し, Kronecker 積 $P \otimes Q : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$ を $P \otimes Q : X \mapsto QXP^T$ と定義する. また, 行列 $X \in \mathbb{R}^{n \times n}$ に対し, $\text{diag}(X) \in \mathbb{R}^n$ で X の対角成分からなるベクトルを表す. 逆に, ベクトル $x \in \mathbb{R}^n$ に対し, $\text{Diag}(x) \in \mathbb{S}^n$ で対角成分が x である対角行列を表す. さらに, 行列 $P \in \mathbb{R}^{m \times n}$ に対し, 線形写像 $\text{Diag}(P) : \mathbb{R}^{m \times n}$ を $\text{Diag}(P) : X \mapsto P \circ X$ と定義する. ここで \circ は Hadamard 積を表す. さらに, 行列 $X \in \mathbb{R}^{m \times n}$ と $p \in \mathbb{R}$ に対し, $X^{(p)} \in \mathbb{R}^{m \times n}$ は (i, j) 成分が X_{ij}^p である行列を表す.

2 面的縮小

この節では (CPO) に対する次の非負半正定値緩和問題について考える:

$$\begin{array}{ll} \text{minimize} & \begin{pmatrix} 0 & \gamma^T \\ \gamma & \Gamma \end{pmatrix} \bullet \begin{pmatrix} \xi & x^T \\ x & X \end{pmatrix} \\ \text{subject to} & \begin{pmatrix} 1 & 0^T \\ 0 & O \end{pmatrix} \bullet \begin{pmatrix} \xi & x^T \\ x & X \end{pmatrix} = 1, \\ & \begin{pmatrix} 0 & \alpha_i^T \\ \alpha_i & O \end{pmatrix} \bullet \begin{pmatrix} \xi & x^T \\ x & X \end{pmatrix} = 2\beta_i \quad (i = 1, \dots, m), \\ & \begin{pmatrix} 0 & 0^T \\ 0 & \alpha_i \alpha_i^T \end{pmatrix} \bullet \begin{pmatrix} \xi & x^T \\ x & X \end{pmatrix} = \beta_i^2 \quad (i = 1, \dots, m), \\ & \begin{pmatrix} 1 & -2e_j^T \\ -2e_j & 4e_j e_j^T \end{pmatrix} \bullet \begin{pmatrix} \xi & x^T \\ x & X \end{pmatrix} = 1 \quad (j \in B), \\ & Y = \begin{pmatrix} \xi & x^T \\ x & X \end{pmatrix} \in S_+^{1+n} \cap S_{(+)}^{1+n}. \end{array} \quad (\text{DNR})$$

ここで $e_j \in \mathbb{R}^n$ は第 j 成分が 1 でそれ以外の成分が 0 のベクトルである. この問題に対し, 正定値かつすべての成分が正であるような実行可能解を実行可能内点解という.

この問題に対し, $R = (-\beta, A)^T$ とおき, $S = RR^T$ とすると, (DNR) の任意の実行可能解 Y に対し, $Y \bullet S = 0$ が成り立つ. これと S の半正定値性より, Y が正定値でないことがわかる. したがって, (DNR) は実行可能内点解をもたない. なお, (DNR) が実行可能内点解をもたないことを示す際には Y の半正定値性のみを用いている. そのため, (CPO) における $(C^{1+n})^*$ を S_+^{1+n} に含まれる錐で置き換えた問題も実行可能内点解をもたないことを示すことができる. また, Burer [7] は (CPO) のサイズを縮小する方法を提案しており, $m = 1$ の場合に実行可能内点解が出現する例を示している. しかし, $m \geq 2$ の場合はこの方法を適用しても実行可能内点解が出現しないことが Tanaka *et al.* [21, Appendix A] により示されている.

Kobayashi *et al.* [13] が指摘しているように, 実行可能内点解をもたない錐最適化問題を主双対内点法で解く場合, 数値的に不安定になる. そこで, この S を用いて (DNR) のサイズを縮小する. この縮小法は Waki and Muramatsu [26, Section 4] で提案された面的縮小に基づく. 錐最適化問題に対する面的縮小は有限回の反復で元の問題と等価で実行可能内点解をもつ問題を出力する. 一般的な面

的縮小についての詳細は [5, 6, 17, 18, 19, 26] を参照されたい. 本論文で提案する縮小法はこの面的縮小の 1 反復に相当するため, 提案する縮小法を適用して得られる非負半正定値緩和問題は実行可能内点解をもつとは限らないが, 主双対内点法の数値的安定性を高めることができると期待できる. そのために $V = (L, R)$ を正則にする $L \in \mathbb{R}^{(1+n) \times (1+n-m)}$ をとる. この L のとり方には自由度があるが, 以下では Tanaka *et al.* [21, Section 2.1] で提案されている L を用いた V を用いて (DNR) のサイズを縮小することを考える. いま, $V^T Y V = (L^T Y L, L^T Y R; R^T Y L, R^T Y R)$ に対し, (DNR) の任意の実行可能解 Y に対して $Y \bullet S = 0$ が成り立つことから, $R^T Y R = O_{m \times m}$ が成り立つ. このことと $V^T Y V$ の半正定値性より, $L^T Y R = O_{(1+n-m) \times m}$ が成り立つ. このことを用いると, (DNR) と等価な次の問題を得る:

$$\begin{array}{ll}
 \text{minimize} & V^{-1} \begin{pmatrix} 0 & \gamma^T \\ \gamma & \Gamma \end{pmatrix} V^{-T} \bullet V^{-1} Y V \\
 \text{subject to} & V^{-1} \begin{pmatrix} 1 & 0^T \\ 0 & O \end{pmatrix} V^{-T} \bullet V^T Y V = 1, \\
 & V^{-1} \begin{pmatrix} 0 & \alpha_i^T \\ \alpha_i & O \end{pmatrix} V^{-T} \bullet V^T Y V = 2\beta_i \quad (i = 1, \dots, m), \\
 & V^{-1} \begin{pmatrix} 0 & 0^T \\ 0 & \alpha_i \alpha_i^T \end{pmatrix} V^{-T} \bullet V^T Y V = \beta_i^2 \quad (i = 1, \dots, m), \\
 & V^{-1} \begin{pmatrix} 1 & -2e_j^T \\ -2e_j & 4e_j e_j^T \end{pmatrix} V^{-T} \bullet V^T Y V = 1 \quad (j \in B), \\
 & V^T Y V = \begin{pmatrix} X & O_{(1+n-m) \times m} \\ O_{m \times (1+n-m)} & O_{m \times m} \end{pmatrix}, \\
 & X \in S_+^{1+n-m}, \quad Y \in S_{(+)}^{1+n}.
 \end{array} \tag{DNR2}$$

さらに, $U = (I_{1+n-m}, O_{(1+n-m) \times m}) V^{-1}$, $C = U(0, \gamma^T; \gamma, \Gamma) U^T$, $A_j = U(1, -2e_j^T; -2e_j, 4e_j e_j^T) U^T$ ($j \in B$) とおくことにより, (DNR2) のサイズを次のように縮小することができる:

$$\begin{array}{ll}
 \text{minimize} & C \bullet X \\
 \text{subject to} & X_{11} = 1, \\
 & A_j \bullet X = 1 \quad (j \in B), \\
 & Y = U^T X U, \\
 & X \in S_+^{1+n-m}, \quad Y \in S_{(+)}^{1+n}.
 \end{array} \tag{FRDNR}$$

ここで (FRDNR) のすべての係数行列のサイズが $(1+n-m) \times (1+n-m)$ と縮小されていることに注意する. また, この縮小により (DNR2) の 2 番目と 3 番目の線形等式制約に対応する線形等式制約が冗長になるため, (FRDNR) ではそれらを削除している. 詳細については Tanaka *et al.* [21, Section 2.1] を参照されたい.

3 主双対パス追跡法

この節では, 前節で述べた面的縮小を適用した非負半正定値最適化問題の主問題:

$$\begin{array}{ll}
 \text{minimize} & C \bullet X \\
 \text{subject to} & \mathcal{A}X = b, \\
 & Y - U^T X U = O, \\
 & X \in S_+^{n-r}, \quad Y \in S_{(+)}^n
 \end{array} \tag{P}$$

とその双対問題:

$$\begin{cases} \text{maximize} & \mathbf{b}^T \mathbf{y} \\ \text{subject to} & \mathcal{A}^T \mathbf{y} + \mathbf{S} + \mathbf{U} \mathbf{T} \mathbf{U}^T = \mathbf{C}, \\ & \mathbf{S} \in \mathcal{S}_+^{n-r}, \quad \mathbf{T} \in \mathcal{S}_{(+)}^n \end{cases} \quad (\text{D})$$

を同時に解く主双対パス追跡法を提案する. ここで, $\mathcal{A}: \mathbf{X} \mapsto (\mathbf{A}_1 \bullet \mathbf{X}, \dots, \mathbf{A}_m \bullet \mathbf{X})^T$ は与えられた線形写像であり, \mathcal{A}^T は \mathcal{A} の随伴写像, すなわち $\mathcal{A}^T: \mathbf{y} \mapsto \sum_{i=1}^m \mathbf{A}_i y_i$ である.

まず, (P) と (D) に対する中心パスを次のように定義する:

$$\left\{ \begin{array}{ll} \mathcal{A} \mathbf{X} = \mathbf{b}, & \mathbf{Y} - \mathbf{U}^T \mathbf{X} \mathbf{U} = \mathbf{O}, \quad \mathbf{X} \in \mathcal{S}_{++}^{n-r}, \quad \mathbf{Y} \in \mathcal{S}_{(++)}^n, \\ (\mathbf{X}, \mathbf{Y}, \mathbf{y}, \mathbf{S}, \mathbf{T}): & \mathcal{A}^T \mathbf{y} + \mathbf{S} + \mathbf{U} \mathbf{T} \mathbf{U}^T = \mathbf{C}, \quad \mathbf{S} \in \mathcal{S}_{++}^{n-r}, \quad \mathbf{T} \in \mathcal{S}_{(++)}^n, \\ & \exists \mu > 0, \quad \mathbf{X} \mathbf{S} = \mu w \mathbf{I}, \quad \mathbf{Y} \circ \mathbf{T} = \mu \mathbf{E} \end{array} \right\}.$$

ここで \mathcal{S}_{++}^n と $\mathcal{S}_{(++)}^n$ はそれぞれ \mathcal{S}_+^n と $\mathcal{S}_{(+)}^n$ の内部を表す. また, $w > 0$ は重み付けパラメータ (定数) である. 中心パスは滑らかな曲線となっており, $\mu \downarrow 0$ のとき最適解の 1 つに収束する. そのため, この曲線を数値的に追跡することにより最適解を得ることができる.

続いて, (P) と (D) に対する主双対パス追跡法の概要を述べる. このアルゴリズムでは次のように中心パスを数値的に追跡する:

(Step 0) 初期点 $(\mathbf{X}, \mathbf{Y}, \mathbf{y}, \mathbf{S}, \mathbf{T}) \in \mathcal{S}_{++}^{n-r} \times \mathcal{S}_{(++)}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^{n-r} \times \mathcal{S}_{(++)}^n$ とパラメータ $\sigma \in (0, 1)$, $\gamma \in (0, 1)$ を定める.

(Step 1) 現在の反復点 $(\mathbf{X}, \mathbf{Y}, \mathbf{y}, \mathbf{S}, \mathbf{T})$ が最適解に十分近ければ終了する.

(Step 2) スケーリングされた Newton 方程式を近似的に解き, 最適解に近い中心パス上の点に近づく探索方向 $(\Delta \mathbf{X}, \Delta \mathbf{Y}, \Delta \mathbf{y}, \Delta \mathbf{S}, \Delta \mathbf{T})$ を求める.

(Step 3) 次の反復点が錐に入るような最大のステップサイズ, すなわち $\alpha_p := \min\{\max\{\alpha : \mathbf{X} + \alpha \Delta \mathbf{X} \in \mathcal{S}_+^{n-r}\}, \max\{\alpha : \mathbf{Y} + \alpha \Delta \mathbf{Y} \in \mathcal{S}_{(+)}^n\}\}$, $\alpha_d := \min\{\max\{\alpha : \mathbf{S} + \alpha \Delta \mathbf{S} \in \mathcal{S}_+^{n-r}\}, \max\{\alpha : \mathbf{T} + \alpha \Delta \mathbf{T} \in \mathcal{S}_{(+)}^n\}\}$ を求める.

(Step 4) ステップサイズを $\alpha_p := \min\{1, \gamma \alpha_p\}$, $\alpha_d := \min\{1, \gamma \alpha_d\}$ とし, $(\mathbf{X}, \mathbf{Y}, \mathbf{y}, \mathbf{S}, \mathbf{T}) := (\mathbf{X} + \alpha_p \Delta \mathbf{X}, \mathbf{Y} + \alpha_p \Delta \mathbf{Y}, \mathbf{y} + \alpha_d \Delta \mathbf{y}, \mathbf{S} + \alpha_d \Delta \mathbf{S}, \mathbf{T} + \alpha_d \Delta \mathbf{T})$ と反復点を更新し, (Step 1) に戻る.

一般的な主双対パス追跡法では, (Step 2) においてスケーリングされた Newton 方程式を縮小した線形方程式系を直接法を用いて正確に解く. しかし今回の場合, 計算量の観点から直接法を適用することは困難である. そのため, (Step 2) ではスケーリングされた Newton 方程式を縮小した拡大方程式系と呼ばれる線形方程式系を解く. この線形方程式系の次元は依然として大きいため, 直接法では現実的な計算量で解くことができない. そのため, 反復法を用いて計算量を削減することを試みる. この方法は, Toh [23] によって提案された凸 2 次半正定値最適化問題に対する主双対パス追跡法に基づく.

本論文では, 前処理の構築しやすさの観点から探索方向として NT (Nesterov-Todd) 方向 [22] を考える. 現在の反復点 $(\mathbf{X}, \mathbf{Y}, \mathbf{y}, \mathbf{S}, \mathbf{T})$ に対し, 探索方向 $(\Delta \mathbf{X}, \Delta \mathbf{Y}, \Delta \mathbf{y}, \Delta \mathbf{S}, \Delta \mathbf{T})$ の計算は次の拡大方程式系の求解に帰着できる:

$$\begin{pmatrix} -\mathcal{H} & \mathcal{A}^T \\ \mathcal{A} & \mathbf{O} \end{pmatrix} \begin{pmatrix} \Delta \mathbf{X} \\ \Delta \mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{R}_a \\ \mathbf{r}_p \end{pmatrix}. \quad (\text{AS})$$

ここで,

$$\begin{aligned}\mathcal{H} &= W^{-1} \otimes W^{-1} + (U \otimes U) \text{Diag}(Y^{(-1)} \circ T)(U \otimes U)^T, \\ R_a &= R_d - R_{\text{psd}} - U(Y^{(-1)} \circ R_{nn})U^T + U(Y^{(-1)} \circ T \circ R_p)U^T\end{aligned}$$

である. \mathcal{H}^{-1} の計算に膨大な計算量を要することから, (AS) から ΔX を消去することは難しい. そのため, (AS) を解くことにより探索方向を計算する. (P) と等価な半正定値最適化問題を一般的な主双対内点法を用いて解く場合, $n(n+1)/2 + m$ 次元の Schur 方程式を解く. 一方, 提案手法で解く (AS) の次元は $(n-r)^2 + m$ と小さくなっていることに注意する.

(AS) を直接法で解くためには, $O[(n-r)^2 + m]^3$ の計算時間と $O[(n-r)^2 + m]^2$ の記憶領域を要する. そのため, 中規模以上の問題を解く際に直接法を用いることは現実的ではない. したがって, (AS) を反復法を用いて解くことを考える. Toh [23] は PSQMR 法 [11] を用いると, (AS) と構造の近い線形方程式系を効率良く解くことができることを示している. そのため, 以下では PSQMR 法を用いることとする.

PSQMR 法は Freund and Nachtigal [11] によって提案された Krylov 部分空間法の 1 つである. 各反復で計算する係数行列とベクトルの積は次のように効率よく計算できる:

$$\begin{pmatrix} -\mathcal{H} & \mathcal{A}^T \\ \mathcal{A} & O \end{pmatrix} \begin{pmatrix} Q \\ q \end{pmatrix} = \begin{pmatrix} -W^{-1}QW^{-1} - U[Y^{(-1)} \circ T \circ (U^TQU)]U^T + \mathcal{A}^Tq \\ \mathcal{A}Q \end{pmatrix}.$$

この右辺は $O(n^2(n-r) + m(n-r)^2)$ の計算時間と $O(n^2 + m)$ の記憶領域で計算することができる. したがって, PSQMR 法の反復回数を抑えることができれば, (AS) を少ない記憶領域で高速に解くことができる.

反復法を用いる欠点は (AS) の残差が 0 にならないことにある. 提案手法では, このことが主双対パス追跡法の収束性に支障をきたさないように Kojima *et al.* [14] が提案している補正を施した. 詳細については Tanaka *et al.* [21, Section 3.1] を参照されたい.

次に, PSQMR 法の収束性を高めるために (AS) に対する 3 つの前処理を提案する. 拡大方程式系の係数行列は, 主双対パス追跡法の反復点が最適解に近づくにつれて悪条件になる. 一般に, Krylov 部分空間法の収束性は係数行列が悪条件であるときに悪化するため, 拡大方程式系の係数行列に対して適切な前処理を行うことが不可欠である. Toh [23] は (AS) と構造の近い線形方程式系に対する 3 つの効果的な前処理を提案している. 以下で提案する前処理 \mathcal{V}^* , \mathcal{P}^* , \mathcal{Z} は, これらの前処理の拡張である.

以下で提案する前処理はすべて $(-\mathcal{K}, \mathcal{A}^T; \mathcal{A}, O)$ という形をもつ. ここで \mathcal{K} は \mathcal{K}^{-1} の計算が容易な \mathcal{H} の近似である. $(R; r) \in S^{n-r} \times \mathbb{R}^m$ に対し, $(Q; q) = (-\mathcal{K}, \mathcal{A}^T; \mathcal{A}, O)^{-1}(R; r)$ は簡単に計算できる. 以下では, \mathcal{K} の構成方法の概要について述べる. 前処理についての詳細は Tanaka *et al.* [21, Section 3.2] を参照されたい.

■**Kronecker 積近似** まず, Kronecker 積近似に基づく方法について述べる. この方法は Toh [23] によって提案された \mathcal{V} の拡張である. Kronecker 積近似とは, 与えられた行列 $G_1, \dots, G_q \in \mathbb{R}^{n \times n}$, $K_1, \dots, K_q \in \mathbb{R}^{m \times m}$ に対し, 次のように q 個の Kronecker 積の和を 1 個の Kronecker 積で近似する

ことである:

$$\min_{V_L \in \mathbb{R}^{n \times n}, V_R \in \mathbb{R}^{m \times m}} \left\| \sum_{p=1}^q G_p \otimes K_p - V_L \otimes V_R \right\|_F^2.$$

Langville and Stewart [15] と Toh *et al.* [25] はこの問題の最適解が効率よく計算できることを示している. この近似を用いるために, \mathcal{H} を Kronecker 積の和で表す. そのために $Y^{(-1)} \circ T$ を次のように固有値分解する:

$$Y^{(-1)} \circ T = \sum_{i=1}^n \lambda_i q_i q_i^T.$$

これらの固有値と固有ベクトルを用いて, 次のように Kronecker 積近似を施す:

$$\mathcal{H} = W^{-1} \otimes W^{-1} + \sum_{i=1}^n \sqrt{\sigma_i \lambda_i} U \text{Diag}(q_i) U^T \otimes \sigma_i \sqrt{\sigma_i \lambda_i} U \text{Diag}(q_i) U^T \simeq V \otimes V = \mathcal{K}.$$

ここで $\sigma = (\text{sgn}(\lambda_1), \dots, \text{sgn}(\lambda_n))^T$ である. なお, $V := V_L^* = V_R^* \in \mathcal{S}_{++}^{n-r}$ を満たす最適な Kronecker 積近似が存在することが Tanaka *et al.* [21, Appendix B] によって示されている.

■**ブロック Kronecker 積近似** 次に, ブロック Kronecker 積近似に基づく方法について述べる. この方法は Toh [23] によって提案された Ψ の拡張である. この前処理では W^{-1} の固有値の大きさに基づいて係数行列の $(1, 1)$ ブロックを分割する. Toh [23] が述べているように, l を最大相補解における X の階数としたとき, W^{-1} は $\Theta(\sqrt{\mu})$ のオーダーをもつ l 個の小さな固有値と, $\Theta(1/\sqrt{\mu})$ のオーダーをもつ $n-r-l$ 個の大きな固有値をもつ. これに基づき W^{-1} を次のように分割する:

$$W^{-1} = P D P^T = P_1 D_1 P_1^T + P_2 D_2 P_2^T.$$

ここで $D_1 = \text{Diag}(d_1) \in \mathcal{S}_{++}^l$ と $D_2 = \text{Diag}(d_2) \in \mathcal{S}_{++}^{n-r-l}$ はそれぞれ W^{-1} の小さな固有値と大きな固有値を対角成分にもつ対角行列であり, $P_1 \in \mathbb{R}^{(n-r) \times l}$ と $P_2 \in \mathbb{R}^{(n-r) \times (n-r-l)}$ はそれぞれ D_1 と D_2 に対応する W^{-1} の固有ベクトルからなる行列である. すなわち $D = \text{Diag}(d_1; d_2)$, $P = (P_1, P_2)$ である. この表記を用いると, \mathcal{H} を次のように表すことができる:

$$\mathcal{H} = (P \otimes P) \left[D \otimes D + \sum_{i=1}^n \sqrt{\sigma_i \lambda_i} P^T U \text{Diag}(q_i) U^T P \otimes \sigma_i \sqrt{\sigma_i \lambda_i} P^T U \text{Diag}(q_i) U^T P \right] (P \otimes P)^T.$$

ここで, 次のようにブロック対角近似を施す:

$$P^T U \text{Diag}(q_i) U^T P \simeq \begin{pmatrix} P_1^T U \text{Diag}(q_i) U^T P_1 & O \\ O & P_2^T U \text{Diag}(q_i) U^T P_2 \end{pmatrix} \quad (i = 1, \dots, n).$$

さらに, 次のように Kronecker 積近似を施す:

$$\widehat{\mathcal{H}}_{ij}^* \simeq D_j \otimes D_i + \sum_{k=1}^n Z_{kj} \otimes \sigma_i Z_{ki} \quad (i, j = 1, 2).$$

ここで,

$$Z_{ki} = \sqrt{\sigma_k \lambda_k} P_i^T U \text{Diag}(q_k) U^T P_i \quad (k = 1, \dots, n; i = 1, 2)$$

である. これらを用いて次のように \mathcal{H} の第 2 因子を近似する:

$$\widehat{\mathcal{H}}^* : \begin{pmatrix} \mathbf{M}_{11} & \mathbf{M}_{21}^T \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{pmatrix} \mapsto \begin{pmatrix} \widehat{\mathcal{H}}_{11}^*(\mathbf{M}_{11}) & [\widehat{\mathcal{H}}_{21}^*(\mathbf{M}_{21})]^T \\ \widehat{\mathcal{H}}_{21}^*(\mathbf{M}_{21}) & \widehat{\mathcal{H}}_{22}^*(\mathbf{M}_{22}) \end{pmatrix}.$$

ここで $\mathbf{M}_{11} \in \mathcal{S}^l$, $\mathbf{M}_{21} \in \mathbb{R}^{(n-r-l) \times l}$, $\mathbf{M}_{22} \in \mathcal{S}^{(n-r-l)}$ である. 最終的に次の近似を得る:

$$\mathcal{K} = (\mathbf{P} \otimes \mathbf{P}) \widehat{\mathcal{H}}^* (\mathbf{P} \otimes \mathbf{P})^T.$$

■**対角スケーリング** 最後に, 一種の対角スケーリングによる方法を述べる. この方法は Toh [23] によって提案された Φ_{\pm} の拡張とみなすことができる. この前処理では \mathbf{W}^{-1} の固有値分解 $\mathbf{W}^{-1} = \mathbf{P} \mathbf{D} \mathbf{P}^T$ を用いて \mathcal{H} を次のように表現する:

$$\mathcal{H} = (\mathbf{P} \otimes \mathbf{P}) \left[\mathbf{D} \otimes \mathbf{D} + (\mathbf{P}^T \mathbf{U} \otimes \mathbf{P}^T \mathbf{U}) \text{Diag}(\mathbf{Y}^{(-1)} \circ \mathbf{T}) (\mathbf{P}^T \mathbf{U} \otimes \mathbf{P}^T \mathbf{U})^T \right] (\mathbf{P} \otimes \mathbf{P})^T.$$

この第 2 因子を次のように対角写像で近似する:

$$\mathbf{D} \otimes \mathbf{D} + (\mathbf{P}^T \mathbf{U} \otimes \mathbf{P}^T \mathbf{U}) \text{Diag}(\mathbf{Y}^{(-1)} \circ \mathbf{T}) (\mathbf{P}^T \mathbf{U} \otimes \mathbf{P}^T \mathbf{U})^T \simeq \text{Diag}[\text{diag}(\mathbf{D}) \text{diag}(\mathbf{D})^T + \mathbf{A}].$$

以上より次の近似を得る:

$$\mathcal{K} = (\mathbf{P} \otimes \mathbf{P}) \text{Diag}[\text{diag}(\mathbf{D}) \text{diag}(\mathbf{D})^T + \mathbf{A}] (\mathbf{P} \otimes \mathbf{P})^T.$$

対角写像近似の簡単な計算方法については Tanaka *et al.* [21, Section 3.2] を参照されたい.

4 数値実験

この節では, 提案手法の効率性を示すために行った数値実験の結果について述べる. そのために, 2 次多次元ナップサック問題とポートフォリオ最適化問題の非負半正定値緩和問題を以下に挙げる 3 つの解法を用いて解き, 比較を行った:

- (S1) (DNR) と等価な半正定値最適化問題を既存のソルバで解く.
- (S2) (FRDNR) と等価な半正定値最適化問題を既存のソルバで解く.
- (S3) (FRDNR) を提案手法で解く.

以下の実験は, OS が Red Hat Enterprise Linux 5.5, CPU が Intel Xeon 2.66 GHz, メモリが 24 GB である計算機上で行った. また, 半正定値最適化問題に対するソルバは SDPA (version 7.3.1) を使い, すべて規定値のパラメータを用いた. なお, 計算時間が 20,000 秒に達した時点で計算を打ち切った.

提案手法は MATLAB (version 7.11) 上で実装した. 予備実験の結果, Ψ^* と Σ は現在の反復点が最適解から遠いときはあまり効果的でないことがわかったため, 以下の実験では次の規則に従って前処理を行った:

- (P1) すべての反復で \mathcal{V}^* を用いる.
- (P2) $\text{err6} > 1$ のとき \mathcal{V}^* を, $\text{err6} \leq 1$ のとき Ψ^* を用いる.
- (P3) $\text{err6} > 1$ のとき \mathcal{V}^* を, $\text{err6} \leq 1$ のとき Σ を用いる.

提案手法の実装に関する詳細は Tanaka *et al.* [21, Section 3.3] を参照されたい。

まず, 2 次多次元ナップサック問題:

$$\begin{array}{ll} \text{maximize} & \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} \\ \text{subject to} & \mathbf{a}_i^T \mathbf{x} \leq b_i \quad (i = 1, \dots, m), \\ & \mathbf{x} \geq \mathbf{0}, \\ & x_j \in \{0, 1\} \quad (j = 1, \dots, n) \end{array} \quad (\text{QMKP})$$

の非負半正定値緩和問題を解いた結果を Table 1 に示す. **err1**, **err3**, **err6** はそれぞれ主問題の残差, 双対問題の残差, 双対ギャップを DIMACS errors [16] で測ったものである. また, **time** は計算時間 (秒), **solver** は使った解法, **precond** は解法として (S3) を用いた場合に用いた前処理を表す. (S2) を用いた場合, $n = 100, m = 20$ の場合を除いて時間切れとなった. これは, 面的縮小により係数行列の疎性が失われたことによると考えられる. $n = 100, m = 20$ の場合, (S2) を用いて得た解の双対ギャップが最も小さいが, 非常に長い計算時間がかかっている. したがって, SDPA は (QMKP) の (FRDNR) と等価な半正定値最適化問題を高速に解くことはできないといえる. 一方, (S1) と (S3) を用いた場合は多くの問題で解を得ることができた. m が小さいとき, (S1) は (S3) よりも高速だった. しかし, m が大きくなるにつれて (S1) の計算時間が増加し, 特に $n = 150, m = 90$ のときは時間切れとなった. 一方, 提案手法を用いた場合の計算時間は m に大きく依存することがなかった. また, 解の精度については (S3) を用いた場合のほうが優れていた.

表 1 (QMKP) に対する非負半正定値緩和問題を 3 つの解法を用いて解いた結果

n	m	solver/precond	(S1)/---	(S2)/---	(S3)/(P1)	(S3)/(P2)	(S3)/(P3)
100	20	err1	1.9e-07	9.9e-04	4.0e-11	4.4e-11	4.1e-11
		err3	2.3e-06	1.3e-12	2.1e-15	1.4e-15	1.8e-15
		err6	5.2e-04	8.3e-06	4.5e-05	4.6e-05	2.0e-05
		time [sec]	197.76	5,767.85	443.39	606.88	403.16
100	60	err1	2.4e-07	---	1.9e-10	1.5e-10	1.6e-10
		err3	3.7e-06	---	9.9e-13	9.9e-13	1.0e-12
		err6	1.5e-03	---	4.4e-05	4.4e-05	9.1e-05
		time [sec]	989.07	> 20,000.00	658.53	704.39	444.73
150	30	err1	2.7e-07	---	5.0e-11	5.7e-11	5.9e-11
		err3	3.5e-06	---	4.8e-15	7.4e-15	7.1e-15
		err6	5.2e-04	---	2.8e-05	4.1e-03	1.7e-05
		time [sec]	1,912.14	> 20,000.00	2,756.81	1,593.80	2,308.51
150	90	err1	---	---	9.3e-10	1.0e-09	1.2e-09
		err3	---	---	3.1e-14	3.0e-14	3.0e-14
		err6	---	---	2.6e-05	4.1e-05	6.3e-05
		time [sec]	> 20,000.00	> 20,000.00	3,207.93	4,119.35	2,532.22

次に, ポートフォリオ最適化問題:

$$\begin{array}{ll} \text{minimize} & \mathbf{x}^T \Sigma \mathbf{x} - 2(\boldsymbol{\mu}^T \mathbf{x})^2 \\ \text{subject to} & \mathbf{e}^T \mathbf{x} = 1, \\ & \mathbf{a}_i^T \mathbf{x} \leq b_i \quad (i = 1, \dots, m), \\ & \mathbf{x} \geq \mathbf{0} \end{array} \quad (\text{Port})$$

の非負半正定値緩和問題を解いた結果を Table 2 に示す. $m = 0$ の場合, すべての解法で同程度の精度の解を得ることができた. 計算時間を比較すると, (S3) は (S1), (S2) に比べて速いといえる. $n = 100, m = 40$ や $n = 100, m = 80$ の場合, (S1) と (S3) では解を得ることができたが (S2) は時間

切れとなった. (S3) を用いて得た解は (S1) を用いて得た解よりも同程度かやや高精度であり, 計算時間を比較しても (S3) は (S1) よりも速い. $n = 150, m = 60$ や $n = 150, m = 120$ の場合は (S3) のみが制限時間内に解を求めることができた. (S3) によって得られた解は十分に高精度である. また, (Port) の (FRDNR) に対する前処理を比較すると, 計算時間の短い (P3) が最も優れた前処理だといえる.

表 2 (Port) に対する非負半正定値緩和問題を 3 つの解法を用いて解いた結果

n	m	solver/precond	(S1)/---	(S2)/---	(S3)/(P1)	(S3)/(P2)	(S3)/(P3)
100	0	err1	4.2e-12	1.1e-10	1.1e-16	4.4e-16	1.7e-16
		err3	2.4e-12	1.1e-12	2.8e-15	2.8e-15	2.8e-15
		err6	8.1e-07	3.9e-07	9.8e-07	9.8e-07	9.8e-07
		time [sec]	78.07	112.85	78.19	62.36	52.23
100	40	err1	2.6e-10	---	2.1e-15	1.8e-15	1.7e-15
		err3	1.0e-10	---	1.6e-14	1.6e-14	1.6e-14
		err6	3.6e-06	---	9.8e-07	4.1e-06	9.4e-07
		time [sec]	774.44	> 20,000.00	175.95	172.37	116.26
100	80	err1	3.7e-09	---	2.6e-16	2.4e-16	2.3e-16
		err3	2.0e-10	---	1.0e-14	1.0e-14	1.0e-14
		err6	9.2e-04	---	9.3e-07	8.3e-07	8.5e-07
		time [sec]	2,247.93	> 20,000.00	324.70	323.09	206.21
150	0	err1	2.1e-12	3.9e-11	2.0e-16	3.4e-16	4.2e-17
		err3	4.0e-12	2.0e-12	3.8e-15	3.8e-15	3.8e-15
		err6	7.5e-07	5.9e-07	6.6e-07	6.6e-07	6.6e-07
		time [sec]	704.51	887.36	325.28	311.83	158.51
150	60	err1	---	---	2.6e-16	2.8e-16	2.6e-16
		err3	---	---	3.3e-14	3.3e-14	3.3e-14
		err6	---	---	5.9e-07	1.2e-06	5.4e-07
		time [sec]	> 20,000.00	> 20,000.00	960.21	978.65	593.90
150	120	err1	---	---	2.8e-16	2.6e-16	2.8e-16
		err3	---	---	2.9e-14	2.9e-14	2.9e-14
		err6	---	---	7.6e-07	8.5e-07	7.2e-07
		time [sec]	> 20,000.00	> 20,000.00	2,436.41	2,634.78	1,528.55

5 おわりに

本論文では 0-1 整数変数を含む非凸 2 次最適化問題 (MBNQO) に対する非負半正定値緩和問題 (DNR) を解く際に現れる次の 2 つの問題点を解決する方法を提案した. 1 つめの問題点は, (DNR) は実行可能内点解をもたないため, 主双対内点法を用いて精度良く解くことが難しいことである. 2 つめの問題点は, 大規模な (DNR) を半正定値最適化問題に帰着させた場合, 問題のサイズが非常に大きくなるため, 既存のソフトウェアで解くことが難しいことである.

本研究では 1 つめの問題点を克服するために問題の退化を用いて (DNR) のサイズを縮小した. この縮小は面的縮小 [26] の 1 反復とみなすことができるため, 主双対内点法の安定性を高めることができる. 面的縮小の適用後も等価な半正定値最適化問題のサイズが大きいという問題点は残る上, 面的縮小により問題の疎性が悪化する場合がある. 本研究ではこの問題点を解決するために縮小後の問題に対する主双対パス追跡法を提案した. この手法は Toh [23] によって提案された凸 2 次半正定値最適化問題に対する主双対パス追跡法に基づく. この手法では探索方向を計算するために大規模な線形方程式系を PSQMR 法 [11] を用いて解いた. また, PSQMR 法の収束性を高めるため, Toh [23] によって提案された前処理を拡張し, 3 つの前処理を提案した. 数値実験の結果, 提案手法

を用いることで非負半正定値緩和問題を速く正確に解くことができることがわかった。

以下に今後の課題を2つ挙げる。面的縮小に関しては、完全正值最適化問題に対して面的縮小を直接適用することにより、より強い下界を得ることができる可能性がある。そのためには、完全正值行列のなす錐の幾何学的構造 [8] などを利用する必要がある。主双対パス追跡法に関しては、係数行列の低ランク性を用いることが考えられる。(FRDNR) の係数行列 \mathcal{A} が低ランク構造をもつことから、SDPT3 [24] で実装されているような計算方法を用いることにより、前処理の計算を高速化することができる。

参考文献

- [1] A. Berman and N. Shaked-Monderer, *Completely Positive Matrices*, World Scientific, Singapore, 2003.
- [2] I. M. Bomze, On standard quadratic optimization problems, *Journal of Global Optimization*, **13** (1998), 369–387.
- [3] I. M. Bomze, Copositive optimization — recent developments and applications, *Technical Report* (2011).
- [4] I. M. Bomze, W. Schachinger and G. Uchida, Think co(mpletely)positive! Matrix properties, examples and a clustered bibliography on copositive optimization, *Technical Report* (2011).
- [5] J. M. Borwein and H. Wolkowicz, Facial reduction for a cone-convex programming problem, *Journal of the Australian Mathematical Society*, **30**, 369–380, 1981.
- [6] J. M. Borwein and H. Wolkowicz, Regularizing the abstract convex program, *Journal of Mathematical Analysis and Applications*, **83**, 495–530, 1981.
- [7] S. Burer, On the copositive representation of binary and continuous nonconvex quadratic programs, *Mathematical Programming*, **120** (2009), 479–495.
- [8] P. J. C. Dickinson, Geometry of the copositive and completely positive cones, *Journal of Mathematical Analysis and Applications*, **380** (2011), 377–395.
- [9] P. J. C. Dickinson and L. Gijben, On the computational complexity of membership problems for the completely positive cone and its dual, *Technical Report*, (2011).
- [10] M. Dür, Copositive programming — a survey. In M. Diehl, F. Glineur, E. Jarlebring and W. Michiels (eds.), *Recent Advances in Optimization and its Applications in Engineering* (pp. 3–20), Springer, New York, 2010.
- [11] R. W. Freund and N. M. Nachtigal, A new Krylov-subspace method for symmetric indefinite linear systems, *Proceedings of the 14th IMACS World Congress on Computational and Applied Mathematics*, 1994, 1253–1256.
- [12] D. Ge and Y. Ye, On doubly positive semidefinite programming relaxations, *Technical Report* (2010).
- [13] K. Kobayashi, K. Nakata and M. Kojima, A conversion of an SDP having free variables into the

- standard form SDP, *Computational Optimization and Applications*, **36** (2007), 289–307.
- [14] M. Kojima, M. Shida and S. Sindoh, Search directions in the SDP and monotone SDLCP: generalization and inexact computation, *Mathematical Programming*, **85** (1999), 51–80.
 - [15] A. N. Langville and W. J. Stewart, A Kronecker product approximate preconditioner for SANs, *Numerical Linear Algebra with Applications*, **11** (2004), 723–752.
 - [16] H. D. Mittelmann, An independent benchmarking of SDP and SOCP solvers, *Mathematical Programming*, **95** (2003), 407–430.
 - [17] G. Pataki, A simple derivation of a facial reduction algorithm and extended dual systems, *Technical Report* (2000).
 - [18] M. V. Ramana, An exact duality theory for semidefinite programming and its complexity implications, *Mathematical Programming*, **77**, 129–162, 1997.
 - [19] M. V. Ramana, L. Tunçel and H. Wolkowicz, Strong duality for semidefinite programming, *SIAM Journal on Optimization*, **7**, 3, 641–662, 1997.
 - [20] J. F. Sturm, Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones, *Optimization Methods and Software*, **11 & 12** (1999), 625–653.
 - [21] M. Tanaka, K. Nakata and H. Waki, Application of a facial reduction algorithm and an inexact primal-dual interior-point method for doubly nonnegative relaxation for mixed binary nonconvex quadratic optimization problems, *Technical Report* (2011).
 - [22] M. J. Todd, K. C. Toh and R. H. Tütüncü, On the Nesterov-Todd direction in semidefinite programming, *SIAM Journal on Optimization*, **8** (1998), 769–796.
 - [23] K. C. Toh, An inexact primal-dual path following algorithm for convex quadratic SDP, *Mathematical Programming*, **112** (2008), 221–254.
 - [24] K. C. Toh, M. J. Todd and R. H. Tütüncü, On the implementation and usage of SDPT3—a MATLAB software package for semidefinite-quadratic-linear programming, version 4.0, *Technical Report* (2010).
 - [25] K. C. Toh, R. H. Tütüncü and M. J. Todd, Inexact primal-dual path-following algorithms for a special class of convex quadratic SDP and related problems, *Pacific Journal of Optimization*, **3** (2007), 135–164.
 - [26] H. Waki and M. Muramatsu, Facial reduction algorithm for conic optimization problems, *Technical Report* (2010).
 - [27] M. Yamashita, K. Fujisawa, K. Nakata, M. Nakata, M. Fukuda, K. Kobayashi and K. Goto, A high-performance software package for semidefinite programs: SDPA 7, *Technical Report* (2010).